



# An Efficient Bit Allocation for Compressing Normal Meshes with an Error-driven Quantization

Frédéric Payan, Marc Antonini

## ► To cite this version:

Frédéric Payan, Marc Antonini. An Efficient Bit Allocation for Compressing Normal Meshes with an Error-driven Quantization. *Computer Aided Geometric Design*, 2005, 22 (5), pp.466–486. 10.1016/j.cagd.2005.04.001 . hal-00264520

**HAL Id: hal-00264520**

**<https://hal.science/hal-00264520>**

Submitted on 17 Mar 2008

**HAL** is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

# An Efficient Bit Allocation for Compressing Normal Meshes with an Error-driven Quantization

Frédéric Payan, Marc Antonini

*Laboratoire I3S - UMR 6070 CNRS  
Université de Nice - Sophia Antipolis  
Route des Lucioles - F-06903 Sophia-Antipolis FRANCE  
Phone: +33 (0)4 92 94 27 22, Fax: +33 (0)4 92 94 28 98  
fpayan@i3s.unice.fr, am@i3s.unice.fr*

---

## Abstract

We propose a new wavelet compression algorithm based on the rate-distortion optimization for densely sampled triangular meshes. This algorithm includes the *normal remesher*, a wavelet transform, and an original bit allocation optimizing the quantization of the wavelet coefficients. The allocation process minimizes the reconstruction error for a given bit budget. As distortion measure, we use the mean square error of the normal mesh quantization, expressed according to the quantization error of each subband. We show that this metric is a suitable criterion to evaluate the reconstruction error, *i.e.* the geometric distance between the input mesh and the quantized *normal* one. Moreover, to design a useful bit allocation, we propose a model-based approach, depending on the wavelet coefficient distributions. The proposed algorithm achieves results better than state-of-the-art methods, up to +2.5 dB compared to the original zerotree coder for normal meshes.

*Key words:* geometry compression, normal meshes, model-based bit allocation, wavelet transform, rate-distortion optimization, multiresolution analysis

*PACS:* Coding

---

## 1 Introduction

Compression algorithm is an essential tool to store, transmit or use 3D objects in bandwidth-limited applications. Although many representations have been proposed for three-dimensional modeling, mesh representation is still omnipresent, and considerable efforts are done in digital geometry processing using primarily triangular meshes [1]. Meshes are a powerful tool to model

complex 3D objects thanks to their double geometrical and combinatorial nature. Indeed, a triangular mesh is represented by a set of vertices (*geometry* information) capturing the global shape of the surface, and a list of triangles (*connectivity* information) describing the incidence relations between the vertices. By consequent, a naïve representation of a highly detailed mesh becomes quickly huge [2] and an efficient mesh compression is essential. Currently, more and more works consider the original mesh to be just one instance of the surface geometry. In that case, we talk about *geometry compression* instead of *mesh compression*. The *geometry compression* considers the geometry to be the most important component to represent a mesh. Therefore geometry coders generally tend to reduce the topology information to the minimum, and one relevant solution is to remesh the irregular input mesh with a *semi-regular remesher* [3,4].

Among the existing schemes of semi-regular remeshing [3–6], *normal meshes* [4] are very attractive because most of vertices lie in a known normal direction, and can be represented by a single scalar (instead of a three-dimensional vector like in [3] for instance). This is currently the most compact multiscale representation of semi-regular meshes. Furthermore, this representation implies that majority of vertices have valence 6. This piecewise sampling regularity allows very efficient wavelet decomposition [7,8], and consequently several wavelet coders exploit them. Let us cite for example the coder NMC proposed by Khodakovsky and Guskov (2002) in [8]. This coder is based on an *unlifted butterfly* wavelet transform and a zerotree coder developed in [7]. By using a local frame depending on the coarser mesh, this transform ensures that for smooth surfaces the majority of wavelet coefficients have no tangential component, and that almost all the geometry information lies in the normal components. The wavelet coefficients are then organized in a multiscale quadtree structure. A zerotree coder followed by an entropy coding is finally applied on each component of the wavelet coefficients, separately. In [9], Jae-Young *et al* (2002) proposed a progressive compression and an interactive transmission algorithm for normal meshes based on rate-distortion optimization. We can also cite the works of Lavu (2003) *et al* [10]. The resulting compression algorithm EQMC is based on an *Estimation-Quantization* framework initially developed for 2D images [11]. This algorithm exploits the spatial and inter-scale correlations of the normal meshes. The authors propose to find the best quantizer for each normal component depending on the normal components previously encoded, in the local neighborhood. This allows to optimize locally the trade-off between the bitrate and the quantization error, providing 0.5-1 dB improvement in coding performance, compared to NMC [8]. In the same way, we proposed in previous works (2003) a bit allocation controlling the quantization error energy to dispatch the bits across wavelet subbands [12,13].

The basic idea of these works is to optimize the trade-off between the bitrate and the quality of the reconstructed mesh either by minimizing the losses

due to the geometry coding, or by reducing the bit budget. This principle called *bit allocation*, is an essential tool to provide an efficient coding when a multiresolution analysis is performed<sup>1</sup>.

In this paper, we propose a model-based bit allocation that optimizes the quantization of the *normal mesh* wavelet coefficients. We focus on the normal meshes because of its simple and multiscale representation. This allows efficient progressive (de)compression and transmission, but also multiresolution analysis and adaptive displaying according to the level-of-detail requirements or hardware capabilities [8–10]. Precisely, we aim to find the best quantizer for each component subband such that the reconstruction error is minimized for a given bitrate. A distortion measure is consequently needed to evaluate the reconstruction error of the quantized mesh.

Several distortion measures have been exploited for compression of irregular meshes [16–19]. For instance, Karni and Gotsman (2000) introduce a metric which captures the visual difference between the original mesh and its approximation [17]. Their criterion depends on the geometric distance and the *laplacian difference* between models. Unfortunately, we cannot use such a *vertex-to-vertex* measure since the proposed coder includes a remeshing technique modifying the topology of the input mesh. In that case, the widely used metric is the symmetric root mean square error between two surfaces [20]. We refer to this error as the *surface-to-surface (S2S) distance*. Based on the *Hausdorff* one, this distance does not depend on the mesh sampling, or its connectivity.

A real computation of the S2S distance is a computationally intensive process. To overcome this problem, we argue that the mean square error relative to the normal mesh quantization, expressed according to the quantization error of each subband, is a suitable criterion to evaluate the reconstruction error between the input mesh and the quantized normal one. Furthermore, this criterion allows to use theoretical models for the bitrate and the distortion of each wavelet subband, involving a fast model-based algorithm of low computational complexity.

We finally design a wavelet coder that includes a bit allocation dispatching the bit budget across the wavelet subbands according to their influence on

---

<sup>1</sup> Note that bit allocation is not only used in case of wavelet coding. Let us cite for instance Chow (1997) or Li and Kuo (1997) who proposed a coder allowing different regions of a mesh to be compressed with different precision in function on the level of details [14,15]. King and Rossignac (1999) focused on the problem of balancing two forms of lossy mesh compression: reduction of the number of vertices by simplification techniques, and reduction of the bitrate per vertex coordinate [16]. More recently, Karni and Gotsman (2000) proposed to truncate spectral coefficients according to a maximum RMS value given as input parameter [17].

the reconstructed mesh quality. Compared to the state-of-the-art coders for normal meshes [8,10], our compression algorithm provides an improvement, in coding performance, up to +2.5 dB at low bitrates.

This paper is organized as follows. In section 2, we introduce some background and notations on triangular meshes and briefly describe the normal meshes. In section 3, we present our framework and the proposed compression algorithm. Then, we deal with a suitable distortion criterion to evaluate the reconstruction error of normal meshes in section 4, and across a wavelet coder in section 5. In section 6 we introduce the proposed bit allocation and develop the model-based algorithm in section 7. Finally, we give some experimental results in section 8, and conclude in section 9.

## 2 Background and notations

**Triangular mesh** Let us denote a triangular mesh  $\mathcal{M}$  as a pair  $(\mathcal{V}, \mathcal{T})$ , where  $\mathcal{V}$  is a set of vertices defined by  $\mathcal{V} = \{v_i = (v_i^x, v_i^y, v_i^z) \in \mathbb{R}^3 / 1 \leq i \leq |\mathcal{V}|\}$  with  $|\mathcal{V}|$  the number of vertices, and  $\mathcal{T}$  a set of triangular faces. Note that a triangular mesh  $\mathcal{M}$  is a piecewise linear surface, and thus can be considered as a continuous surface.

**Semi-regular mesh** A semi-regular triangular mesh  $\mathcal{M}_{sr}$  is a multiscale mesh, built by repeated regular subdivision of a base mesh  $\mathcal{M}_0 = (\mathcal{V}_0, \mathcal{T}_0)$  (a coarse version of the original irregular mesh obtained by a simplification technique [21]), providing several meshes  $\mathcal{M}_i$  ( $\mathcal{M}_1 = (\mathcal{V}_1, \mathcal{T}_1)$ ,  $\mathcal{M}_2 = (\mathcal{V}_2, \mathcal{T}_2)$ , ...) until the finest semi-regular mesh  $\mathcal{M}_{sr} = (\mathcal{V}_{sr}, \mathcal{T}_{sr})$ . These meshes have the notable property:

$$\mathcal{V}_0 \subset \mathcal{V}_1 \dots \subset \mathcal{V}_{sr}.$$

Fig. 1 shows an example of a semi-regular mesh at different resolution levels. The vertices added to obtain a finer mesh can be defined by a set of three-dimensional detail vectors  $\mathcal{D}_i = \{d_{i,j} = (d_{i,j}^x, d_{i,j}^y, d_{i,j}^z) \in \mathbb{R}^3 / 1 \leq j \leq |\mathcal{D}_i|\}$ , with  $|\mathcal{D}_i|$  the number of details at the resolution level  $i$ . The set of details  $\mathcal{D}_i$  describes the deformations between the mesh  $\mathcal{M}_{i-1}$  and  $\mathcal{M}_i$ . The details are mostly computed in a local frame [22] induced by the tangent plane and the normal direction at the surface defined by the mesh of lower resolution [7]. This involves the distinction between the so-called *tangential components* and *normal components* of detail vectors  $d_{i,j}$ :

- the *tangential components* are the coordinates  $d_{i,j}^x$  and  $d_{i,j}^y$  of detail vectors;
- the *normal components* are the coordinates  $d_{i,j}^z$  of detail vectors.

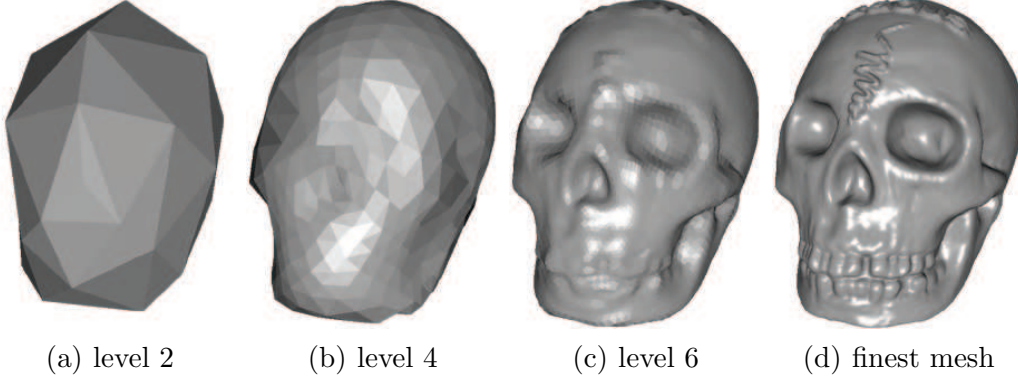


Fig. 1. Multiresolution semi-regular version of SKULL at different levels of resolution, from level 2 (64 triangles) to level 8 (finest mesh with 262144 triangles).

**Normal mesh** Normal meshes are very attractive because majority of the details may be represented with a single number instead of a three-dimensional vector like in [3]. These multiresolution meshes have the property that the details almost always lie in a known normal direction (see Fig. 2) [4]. This involves that the tangential components tends to be equal to zero. This is currently the most compact representation of semi-regular meshes.

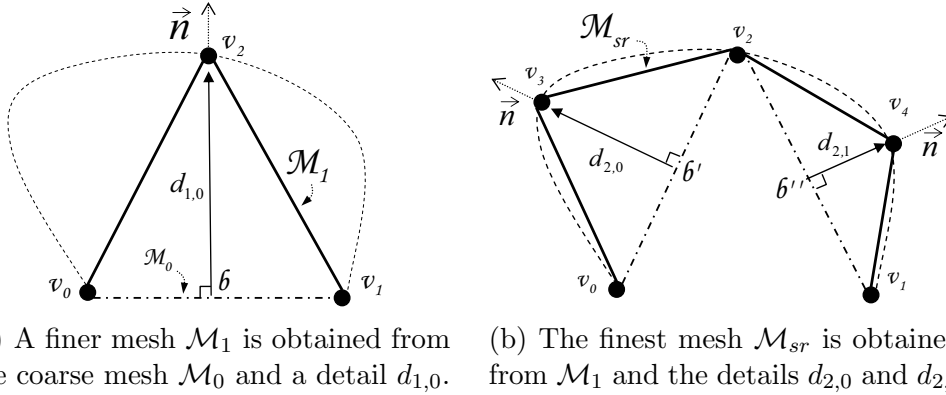


Fig. 2. A *normal mesh*  $\mathcal{M}_{sr}$  is obtained by successive connectivity subdivision of a coarse mesh  $\mathcal{M}_0$ . The detail vectors depends on the normals at the surface.

**Quantized mesh:** let us denote a *quantized normal mesh*  $\hat{\mathcal{M}}_{sr}$  as a pair  $\hat{\mathcal{M}}_{sr} = (\hat{\mathcal{V}}_{sr}, \mathcal{T}_{sr})$ .  $\hat{\mathcal{V}}_{sr}$  represents the set of quantized vertices defined by  $\hat{\mathcal{V}}_{sr} = \{\hat{v}_i \in \mathbb{R}^3 / 1 \leq i \leq |\hat{\mathcal{V}}_{sr}|\}$ , where  $\hat{v}_i = Q(v_i)$ .  $Q(\cdot)$  is called the quantization operator associated to a quantization step  $q$ .

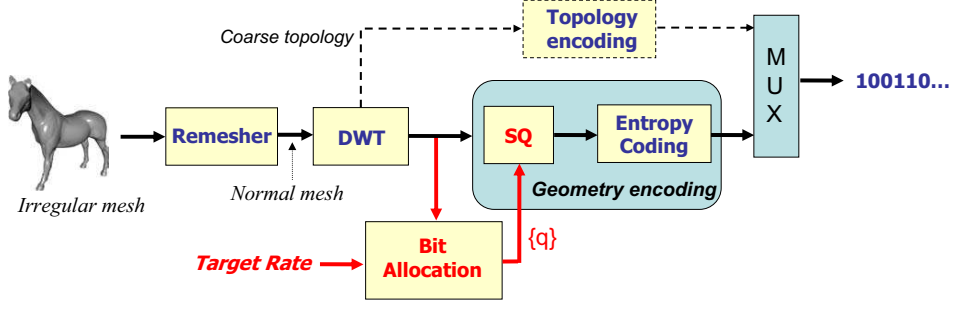


Fig. 3. Proposed geometry coder.

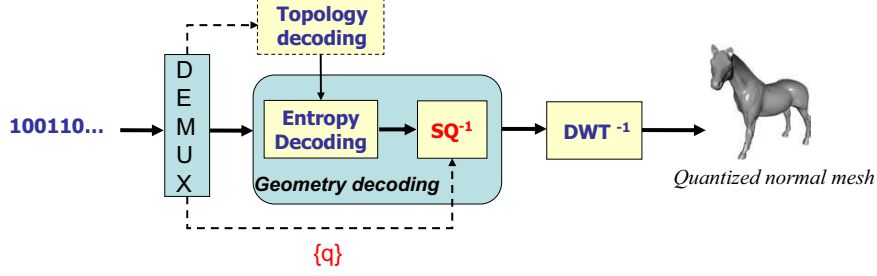


Fig. 4. Proposed geometry decoder.

### 3 Overview of the proposed approach

Fig. 3 and 4 present the global scheme of the proposed coder/decoder. The algorithm principle is described hereinafter. The normal remesher provides a semi-regular mesh  $\mathcal{M}_{sr}$ , from the irregular input one  $\mathcal{M}_{ir}$ . A  $N$ -level unlifted butterfly wavelet transform [23,24] is then applied to obtain  $N$  subbands of three-dimensional wavelet coefficients. Using this wavelet transform ensures that wavelet coefficients remain in the normal direction [8].

The *tangential* and *normal sets* (see section 2) of wavelet coefficients are then encoded separately using uniform scalar quantizers  $SQ$  depending on the optimal quantization steps computed during the allocation process. An entropy coder adapted to the multiresolution semi-regular mesh [13] is finally applied. In parallel, the connectivity of the coarse mesh can be encoded with any topological coder. In this paper, we choose the performant coder of Touma and Gotsman [25]. Finally, the two bitstreams are merged for transmission.

The goal of this paper is to propose a coder/decoder including a bit allocation process that optimizes the quality of the *quantized normal mesh*. A suitable distortion measure  $D_T$  is thus needed to evaluate the reconstruction error during the geometry encoding. In the next section, we deal with the choice of the distortion measure.

## 4 Choice of the distortion measure

### 4.1 The S2S distance as quality criterion

Several distortion measures have been exploited by single-rate mesh coders [16–19]. In this paper, we choose as reconstruction error the symmetric root mean square error between two surfaces [20] also called the S2S distance. We choose this distance because it is generally used to evaluate the performances of coders based on remeshing [7–10]. Indeed, this distance does not depend on the mesh sampling, or connectivity.

The distortion measure  $D_T$  is defined as the energy of the S2S distance between the irregular input mesh  $\mathcal{M}_{ir}$  and the *quantized normal mesh*  $\hat{\mathcal{M}}_{sr}$ :

$$D_T = d_S(\mathcal{M}_{ir}, \hat{\mathcal{M}}_{sr})^2, \quad (1)$$

where  $d_S(.,.)$  represents the S2S distance.

### 4.2 Definition of the S2S distance

The S2S distance between the two meshes  $\mathcal{M}_{ir}$  and  $\hat{\mathcal{M}}_{sr}$  [20] is defined by

$$d_S(\mathcal{M}_{ir}, \hat{\mathcal{M}}_{sr}) = \max[\bar{d}(\mathcal{M}_{ir}, \hat{\mathcal{M}}_{sr}); \bar{d}(\hat{\mathcal{M}}_{sr}, \mathcal{M}_{ir})], \quad (2)$$

where  $\bar{d}(\mathcal{M}, \mathcal{M}')$  is the *unilateral distance* between two meshes [20], given by

$$\bar{d}(\mathcal{M}, \mathcal{M}') = \left( \frac{1}{|\mathcal{M}|} \iint_{p \in \mathcal{M}} d(p, \mathcal{M}')^2 d\mathcal{M} \right)^{\frac{1}{2}}. \quad (3)$$

$|\mathcal{M}|$  represents the area of  $\mathcal{M}$ , and  $d(p, \mathcal{M}')$  represents the distance between a point  $p$  belonging to a surface represented by a mesh  $\mathcal{M}$  and the surface represented by a mesh  $\mathcal{M}'$ . This distance is defined by

$$d(p, \mathcal{M}') = \min_{p' \in \mathcal{M}'} \|p - p'\|_2 = \|p - \text{Proj}_{\mathcal{M}'}(p)\|_2, \quad (4)$$

with  $\|\cdot\|_2$  the  $L_2$ -norm, and  $\text{Proj}_{\mathcal{M}'}(p)$  the orthogonal projection of  $p$  over  $\mathcal{M}'$ .

To avoid a real computation of the S2S distance during the bit allocation, which is a computationally intensive process, we propose to use a simpler but suitable criterion to evaluate the reconstruction error. To this purpose, we make several assumptions.



#### 4.3 First assumption: an optimal remeshing

Notice that the normal remesher provides a *normal mesh*  $\mathcal{M}_{sr}$  very close to the original irregular mesh  $\mathcal{M}_{ir}$ . Table 1 shows that the S2S distance between these two meshes is negligible (lower than 0.016% of the bounding box diagonal).

Model	$ \mathcal{V}_{ir} $	base	refinement level	Remeshing error
FELINE	49919	504	7	0.0090 %
HORSE	48485	220	6	0.0036 %
RABBIT	67039	80	7	0.0042 %
SKULL	20002	4	8	0.0157 %
VENUS	50002	76	6	0.0058 %

Table 1

Remeshing error between the irregular mesh and the normal mesh (S2S distance relative to the bounding box diagonal). “Base” represents the number of triangles of the base mesh.

Eq. (1) can then be approximated by

$$D_T \simeq d_S(\mathcal{M}_{sr}, \hat{\mathcal{M}}_{sr})^2 = \max[\bar{d}(\mathcal{M}_{sr}, \hat{\mathcal{M}}_{sr})^2; \bar{d}(\hat{\mathcal{M}}_{sr}, \mathcal{M}_{sr})^2]. \quad (5)$$

#### 4.4 Second assumption: densely sampled meshes

Let us study the difference of “symmetry” between the distances  $\bar{d}(\mathcal{M}_{sr}, \hat{\mathcal{M}}_{sr})$  and  $\bar{d}(\hat{\mathcal{M}}_{sr}, \mathcal{M}_{sr})$ . Table 2 presents a mean of the relative errors between these two distances, computed on 5 typical models (HORSE, RABBIT, VENUS, SKULL and FELINE), and according to different bitrate ranges. The difference being very low ( $< 4\%$ ) for each bitrate range, we can assume that  $\bar{d}(\mathcal{M}_{sr}, \hat{\mathcal{M}}_{sr}) \simeq \bar{d}(\hat{\mathcal{M}}_{sr}, \mathcal{M}_{sr})$ , and we can simplify the computation of  $D_T$  by using only one of the unilateral distances:

$$D_T \simeq \bar{d}(\hat{\mathcal{M}}_{sr}, \mathcal{M}_{sr})^2, \quad (6)$$

or equivalently,

$$D_T \simeq \frac{1}{|\hat{\mathcal{M}}_{sr}|} \iint_{p \in \hat{\mathcal{M}}_{sr}} d(p, \mathcal{M}_{sr})^2 d\hat{\mathcal{M}}_{sr}. \quad (7)$$

$D_T$  should be computed analytically in each point  $p \in \hat{\mathcal{M}}_{sr}$ . However, since a *normal mesh* is densely sampled, the number of vertices is large. Thus, we can assume a uniform distribution of the vertices on the surface. Consequently,

Bitrate (bits/iv)	< 1	1-2	2-6	6-10	> 10
$\bar{d}(\mathcal{M}_{sr}, \hat{\mathcal{M}}_{sr})$	1.07e-1	1.30e-2	5.34e-3	1.72e-3	1.43e-3
$\bar{d}(\hat{\mathcal{M}}_{sr}, \mathcal{M}_{sr})$	1.02e-1	1.28e-2	5.34e-3	1.72e-3	1.43e-3
Difference (%)	3.680	1.890	0.291	0.261	0.074

Table 2

Mean differences between  $\bar{d}(\mathcal{M}_{sr}, \hat{\mathcal{M}}_{sr})$  and  $\bar{d}(\hat{\mathcal{M}}_{sr}, \mathcal{M}_{sr})$  according to the bitrate per irregular vertex (bits/iv), computed on 5 typical models (HORSE, RABBIT, VENUS, SKULL and FELINE).

the integral in (7) can be numerically approximated by a discrete sum [26]. Moreover, the area of triangles being very small relative to global surface, the distance point-surface  $d(p, \mathcal{M}_{sr})$  can be computed only from the vertices [27]:

$$D_T \simeq \frac{1}{|\hat{\mathcal{V}}_{sr}|} \sum_{\hat{v} \in \hat{\mathcal{V}}_{sr}} d(\hat{v}, \mathcal{M}_{sr})^2, \quad (8)$$

with  $\hat{v} = Q(v)$  the quantized version of the vertex  $v$ , and  $|\hat{\mathcal{V}}_{sr}|$  the number of vertices of  $\hat{\mathcal{M}}_{sr}$  (or equivalently  $\mathcal{M}_{sr}$ ).

Now, we have to deal with  $d(\hat{v}, \mathcal{M}_{sr}) = \|\hat{v} - \text{Proj}_{\mathcal{M}_{sr}}(\hat{v})\|_2$ .

#### 4.5 Third assumption: an optimal bitrate coding

Let us introduce the quantization error vector  $\varepsilon(v) = v - \hat{v}$ , between a vertex  $v$  and its quantized version  $\hat{v}$ . Under the assumption of an optimal bitrate coding and in the considered bitrate range, we can assume that the quantization of the coarser levels does not modify consequently the computation of the local coordinate systems in which the details of finer levels are expressed. This introduces further tangential components, but these components remain small compared to normal components. As a result, and since we use a normal remesher, most of error vectors  $\varepsilon(v)$  lie in the normal direction at the surface  $\mathcal{M}_{sr}$  in  $v$ . Therefore, during the computation of  $d(\hat{v}, \mathcal{M}_{sr}) = \|\hat{v} - \text{Proj}_{\mathcal{M}_{sr}}(\hat{v})\|_2$ , the orthogonal projection of  $\hat{v}$  over  $\mathcal{M}_{sr}$  remains very close to  $v$ :

$$\text{Proj}_{\mathcal{M}_{sr}}(\hat{v}) \simeq v.$$

Finally, we can state that

$$d(\hat{v}, \mathcal{M}_{sr}) = \|\hat{v} - \text{Proj}_{\mathcal{M}_{sr}}(\hat{v})\|_2 \simeq \|\hat{v} - v\|_2 = \|Q(v) - v\|_2. \quad (9)$$

Using Eq. (9), Eq. (8) can be written as

$$D_T \simeq \frac{1}{|\mathcal{V}_{sr}|} \sum_{v \in \mathcal{V}_{sr}} \|Q(v) - v\|_2^2. \quad (10)$$

We notice that the right-hand side of (10) corresponds to the quantization error of the normal mesh geometry, *i.e.*, the MSE denoted by  $\sigma_{Q_{sr}}^2$ . Thus, in case of densely sampled meshes and under the assumption of an optimal bitrate coding, the MSE of the geometry quantization should be a suitable distortion criterion to evaluate the reconstruction error between the irregular input mesh and the quantized one. Finally, we can write

$$\boxed{D_T \simeq \sigma_{Q_{sr}}^2.} \quad (11)$$

This formulation is computed in the euclidean space, and depends on the vertices of the mesh. Now, the proposed bit allocation is processed on the wavelet coefficient subbands. Thus, we have to express the MSE of quantization of the *normal mesh* geometry according to the quantized coefficients.

## 5 MSE across a wavelet coder

We have shown in section 4 that the MSE of normal mesh quantization is a suitable criterion to evaluate the reconstruction error between the irregular input mesh and the quantized normal one. In the proposed coder, the unlifted butterfly wavelet transform is applied on the normal mesh  $\mathcal{M}_{sr}$ . Hence, we obtain the coarse base mesh  $\mathcal{M}_0$ , and  $N$  three-dimensional wavelet coefficient subbands. The geometry  $\mathcal{V}_0$  of the coarse mesh  $\mathcal{M}_0$  is called the *low frequency* subband. The sets  $\mathcal{D}_i$  defined in section 2 are now the *high frequency* or wavelet coefficients subbands, with  $i$  the resolution level.

In [28,12], it is shown that the MSE of a multidimensional signal encoded across a wavelet coder using a  $N$ -level decomposition is equivalent to a weighted sum of the MSE  $\sigma_{Q_i}^2$  introduced by the quantization of each wavelet coefficient subband  $i$ . Therefore, the MSE  $\sigma_{Q_{sr}}^2$  between a normal mesh and its quantized version can be written as

$$\sigma_{Q_{sr}}^2 = \sum_{i=0}^N w_i \sigma_{Q_i}^2, \quad (12)$$

where  $\sigma_{Q_i}^2$  is the MSE due to the quantization of the wavelet coefficient subband  $i$ , and  $\{w_i\}$  are the weights due to the biorthogonality of the wavelet transform. In [12,29], the weights are computed for the lifted butterfly wavelet transform. Similarly, we can compute the weights for the unlifted butterfly wavelet transform:

$$w_i = \frac{|\mathcal{D}_i|}{|\mathcal{V}_{sr}|} \left( \frac{169}{256} \right)^{N-i}, \quad (13)$$

where  $|\mathcal{D}_i|$  is the number of coefficients of the subband  $\mathcal{D}_i$ , and  $|\mathcal{V}_{sr}|$  the number of semi-regular vertices.

Recall that in our framework each subband of high frequency wavelet coefficients is splitted in two scalar sets, the *tangential* and *normal sets* (see section

2). Consequently, the MSE  $\sigma_{Q_i}^2$  of the  $i^{th}$  high frequency subband ( $\forall i \neq 0$ ) is the sum of the MSE  $\sigma_{Q_{i,1}}^2$  and  $\sigma_{Q_{i,2}}^2$  due to the quantization of the tangential and normal sets:

$$\sigma_{Q_i}^2 = \sum_{j \in J_i} \sigma_{Q_{i,j}}^2 \quad \forall i \neq 0, \quad (14)$$

where  $J_i$  is a set of indices defined by  $J_i = \{1, 2\}, \forall i \neq 0$ .

On the other hand, the low frequency subband do not present specific properties, since it represents a coarse version of the input mesh. Therefore, the low frequency subband will be splitted in three scalar sets, and the MSE  $\sigma_{Q_0}^2$  of the low frequency subband is the sum of the three MSE  $\sigma_{Q_{0,j}}^2$  due to the quantization on each coordinate set:

$$\sigma_{Q_0}^2 = \sum_{j \in J_0} \sigma_{Q_{0,j}}^2, \quad (15)$$

where  $J_0$  is a set of coordinate indices defined by  $J_0 = \{1, 2, 3\}$ .

Finally, by merging (14) and (15) in (12), the MSE  $\sigma_{Q_{sr}}^2$  relative to the geometry of a semi-regular mesh encoded with a wavelet coder is given by

$$\sigma_{Q_{sr}}^2 = \sum_{i=0}^N w_i \sum_{j \in J_i} \sigma_{Q_{i,j}}^2 \quad (16)$$

with  $w_i$  given by (13). The formulation (16) is finally used as distortion measure during the bit allocation process to evaluate the distortion introduced on the reconstructed mesh by the geometry quantization.

## 6 Optimal bit allocation

### 6.1 General Purpose

The general purpose of the proposed bit allocation is to optimize the trade-off between the global bitrate and the quality of the reconstructed mesh by controlling and minimizing the losses due to the geometry quantization at a given bitrate. Compared to the algorithm proposed in [10] that optimizes *locally* the trade-off between the bitrate and the quantization error of the coefficients, the proposed bit allocation process aims to *determine the best set of quantization steps  $\{q_{i,j}\}$  used to quantize the subbands, that minimizes the global reconstruction error  $D_T$  of the decoded mesh at a given bitrate  $R_{target}$* . The quantity  $R_{target}$  corresponds to the aimed bitrate for the compressed mesh, expressed here in bits per semi-regular vertex. It can be fixed by either the user, or automatically by the computer, depending on the application or the

bandwidth limitation. The principle is the following. The wanted bitrate is given, and then the reconstruction error is minimized for this specific bitrate. Once the allocation processed and the quantization steps computed for this bitrate, the compression, the transmission and the decompression are done progressively.

The optimization problem can be formulated as follows:

$$(\mathcal{P}) \begin{cases} \text{minimize} & D_T(\{q_{i,j}\}) \\ \text{with constraint} & R_T(\{q_{i,j}\}) = R_{target}, \end{cases} \quad (17)$$

where  $R_T$  is the given total bitrate. By using a lagrangian operator, this constrained allocation problem can be defined by a lagrangian criterion :

$$J_\lambda(\{q_{i,j}\}) = D_T(\{q_{i,j}\}) + \lambda(R_T(\{q_{i,j}\}) - R_{target}), \quad (18)$$

with  $\lambda$  the lagrangian operator. By merging the distortion measure (16) proposed in section 5 with (18), the lagrangian criterion can be developed in:

$$J_\lambda(\{q_{i,j}\}) = \sum_{i=0}^N w_i \sum_{j \in J_i} \sigma_{Q_{i,j}}^2(q_{i,j}) + \lambda \left( \sum_{i=0}^N \sum_{j \in J_i} a_{i,j} R_{i,j}(q_{i,j}) - R_{target} \right), \quad (19)$$

where  $\sigma_{Q_{i,j}}^2(q_{i,j})$  and  $R_{i,j}$  are respectively the MSE and the bitrate relative to the  $(i, j)^{th}$  component set. The coefficients  $a_{i,j}$  depend on the subsampling and correspond to the ratio between the size of the  $(i, j)^{th}$  component set and the total number of components ( $3 \times |\mathcal{V}_{sr}|$ ).

## 6.2 Optimal Solution

The solution of this constrained allocation problem can be obtained by differentiating Eq. (19) with respect to the quantization steps  $\{q_{i,j}\}$  and  $\lambda$  (first order conditions), or equivalently by solving the following system:

$$\begin{cases} \frac{\partial J_\lambda(\{q_{i,j}\})}{\partial q_{i,j}} = 0 \\ \frac{\partial J_\lambda(\{q_{i,j}\})}{\partial \lambda} = 0 \end{cases} \quad (20)$$

This system can be developed in

$$w_i \frac{\partial \sigma_{Q_{i,j}}^2(q_{i,j})}{\partial q_{i,j}} + \lambda a_{i,j} \frac{\partial R_{i,j}(q_{i,j})}{\partial q_{i,j}} = 0 \quad (21a)$$

$$\sum_{i=0}^N \sum_{j \in J_i} a_{i,j} R_{i,j}(q_{i,j}) = R_{target} \quad (21b)$$

Finally, we have to solve the following system of  $(2N + 4)$  equations with  $(2N + 4)$  unknowns (the set  $\{q_{i,j}\}$  and  $\lambda$ ):

$$h_\alpha(q_{i,j}) = \frac{\frac{\partial \sigma_{Q_{i,j}}^2(q_{i,j})}{\partial q_{i,j}}}{\frac{\partial R_{i,j}(q_{i,j})}{\partial q_{i,j}}} = -\lambda \frac{a_{i,j}}{w_i} \quad (22a)$$

$$\sum_{i=0}^N \sum_{j \in J_i} a_{i,j} R_{i,j}(q_{i,j}) = R_{target} \quad (22b)$$

In order to obtain the optimal quantization steps analytically, (22a) requires to be inverted. Unfortunately, this stage is impossible due to the complexity of the equations. To overcome this problem, an iterative algorithm depending on  $\lambda$  is generally proposed.

### 6.3 Overall Algorithm

The optimal solutions of system (22) are then computed thanks to the following overall algorithm:

- (1)  $\lambda$  is given. For each set  $(i, j)$ , compute  $q_{i,j}$  that verifies (22a);
- (2) while (22b) is not verified, calculate a new  $\lambda$  by dichotomy and return to step 1;
- (3) stop.

The computation of the quantization steps  $\{q_{i,j}\}$  as solutions of (22a) can be done according to different methods. In the following section 7, we propose to process this algorithm with an efficient analytical approach thanks to theoretical models for the bitrate and the MSE [30].

## 7 Model-based Approach

The only way to compute the bitrate and the MSE of the different component sets of the wavelet subbands without real pre-quantizations is to perform a model-based bit allocation. Therefore, we introduce theoretical models for the distortion and the bitrate, depending on the probability density functions of each data set. Let us focus now on the estimation of these density functions.

### 7.1 Wavelet coefficients distribution

Fig. 5 shows typical probability density functions of the tangential and normal sets of wavelet coefficients of normal meshes obtained by the unlifted butterfly wavelet transform.

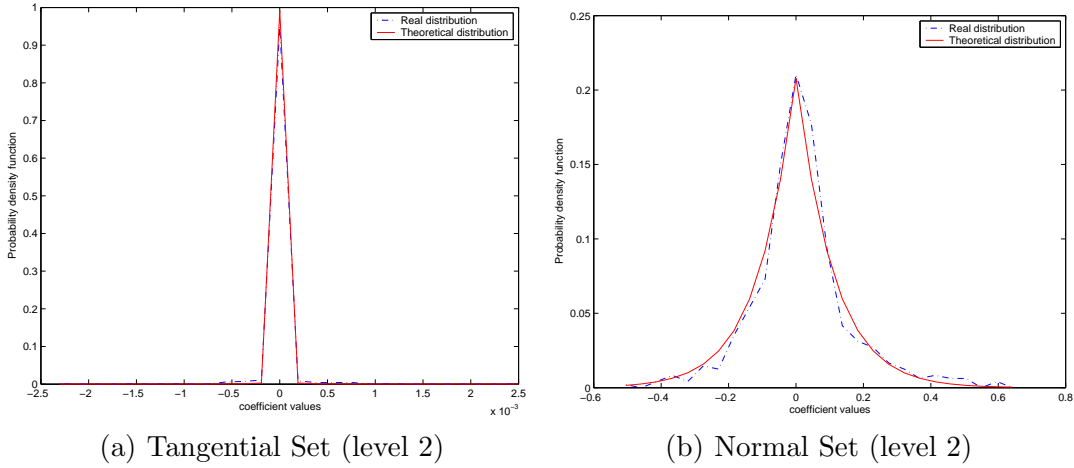


Fig. 5. Typical probability density functions of tangential and normal sets (model *Venus*). The dash-dot lines represent the real density functions, and the solid lines represent the corresponding GGD.

We observe that distributions are zero mean and all informations are concentrated on few coefficients (very small variances). By using a  $\chi^2$ -test, we observe each probability density function of the tangential and normal sets can be modeled by a *Generalized Gaussian Distribution* (GGD)<sup>2</sup>.

The formulation of a GGD is given by

$$p_{\sigma,\alpha}(x) = ae^{-|bx|^\alpha} \quad (23)$$

<sup>2</sup> For instance, the  $\chi^2$ -test applied on the two probability density functions shown in Fig. 5 provides  $\chi^2 = 1.5621$  for the tangential set, and  $\chi^2 = 0.1286$  for the normal set when 26 quantization cells are used, meaning that than 99% of the coefficients are well modeled by a GGD.

with  $b = \frac{1}{\sigma} \sqrt{\frac{\Gamma(3/\alpha)}{\Gamma(1/\alpha)}}$  and  $a = \frac{b\alpha}{2\Gamma(1/\alpha)}$ . The parameter  $\alpha$  is computed using the variance  $\sigma^2$  and the fourth-order moment of each set [31]. Fig. 5 also shows the GGD used to model the real distribution (solid lines).

### 7.2 Processing of the low frequency subband

On the other hand, the three subsets of the low frequency subband do not have any particular distribution and cannot be modeled by an unimodal function like the high frequency component sets, since they represent a coarse version of the original mesh. To overcome this problem, we choose to model and encode the differences between two low frequency components, instead of the components themselves (differential coding) [32]. We observe that these differences can also be modeled by a GGD. However, this method is interesting if no side information is required by the decoder to reconstruct the good connectivity. This is possible if the differential coding is processed by following the ordered list of low frequency vertices given by the topological coder.

### 7.3 Theoretical models for the distortion and the bitrate

The theoretical model to compute the MSE  $\sigma_Q^2$  of a uniform scalar quantizer is given by:

$$\sigma_Q^2 = \int_{-\frac{q}{2}}^{+\frac{q}{2}} p_{\sigma,\alpha}(x) dx + 2 \sum_{m=1}^{+\infty} \int_{\frac{q}{2}+|m|q}^{\frac{q}{2}+(|m|-1)q} (x - \hat{x})^2 p_{\sigma,\alpha}(x) dx, \quad (24)$$

where  $x$  is an original sample and  $\hat{x}$  its decoding value. Furthermore, since an entropy coder is used after the quantization, we suppose that the bitrate  $R$  after encoding is equal to the entropy of the quantized components of each set:

$$R = - \sum_{m=-\infty}^{+\infty} Pr(m) \log_2 Pr(m). \quad (25)$$

$Pr(m)$  is the probability of a quantization level  $m$ :

$$Pr(m) = \int_{\frac{q}{2}+|m|q}^{\frac{q}{2}+(|m|-1)q} p_{\sigma,\alpha}(x) dx \text{ and } Pr(0) = \int_{-\frac{q}{2}}^{+\frac{q}{2}} p_{\sigma,\alpha}(x) dx. \quad (26)$$

where  $p_{\sigma,\alpha}(x)$  is the probability density function of a subset.

Moreover, the authors of [30] (2003) show that for an uniform scalar quantization using the center of the cells as decoding value, the MSE (24) for a GGD



$p_{\sigma,\alpha}(x)$  can be rewritten as

$$\sigma_Q^2 = \sigma^2 D(\tilde{q}, \alpha), \quad (27)$$

with  $\sigma^2$  the variance of the set, and  $\tilde{q} = \frac{q}{\sigma}$ .  $D(\tilde{q}, \alpha)$  is a simple function given by

$$D(\tilde{q}, \alpha) = 1 + 2 \sum_{m=1}^{+\infty} (m\tilde{q})^2 f_{0,m}(\tilde{q}, \alpha) - 4 \sum_{m=1}^{+\infty} m\tilde{q} f_{1,m}(\tilde{q}, \alpha), \quad (28)$$

where functions  $f_{n,m}$  are defined by

$$f_{n,m}(\tilde{q}, \alpha) = \int_{\frac{1}{2}\tilde{q}+(m-1)\tilde{q}}^{\frac{1}{2}\tilde{q}+m\tilde{q}} x^n p_{1,\alpha}(x) dx, \quad (29)$$

and by

$$f_{n,0}(\tilde{q}, \alpha) = \int_{-\frac{1}{2}\tilde{q}}^{\frac{1}{2}\tilde{q}} x^n p_{1,\alpha}(x) dx. \quad (30)$$

By the same way, the bitrate  $R$  associated to a GGD can be rewritten as [30]

$$R(\tilde{q}, \alpha) = -f_{0,0}(\tilde{q}, \alpha) \log_2 f_{0,0}(\tilde{q}, \alpha) - 2 \sum_{m=1}^{+\infty} f_{0,m}(\tilde{q}, \alpha) \log_2 f_{0,m}(\tilde{q}, \alpha). \quad (31)$$

According to the theoretical model (28) for the MSE, and the theoretical model (31) for the bitrate of a component set, the system (22) becomes

$$h_\alpha(\tilde{q}_{i,j}) = \frac{\frac{\partial D(\tilde{q}_{i,j}, \alpha)}{\partial \tilde{q}_{i,j}}}{\frac{\partial R_{i,j}(\tilde{q}_{i,j}, \alpha)}{\partial \tilde{q}_{i,j}}} = -\lambda \frac{a_{i,j}}{w_i \sigma_{i,j}^2} \quad (32a)$$

$$\sum_{i=0}^N \sum_{j \in J_i} a_{i,j} R_{i,j}(\tilde{q}_{i,j}, \alpha) = R_{target} \quad (32b)$$

where  $h_\alpha(\tilde{q}_{i,j})$  can be developed in

$$h_\alpha(\tilde{q}_{i,j}) = \frac{\sum_{m=1}^{+\infty} m \left[ 2f_{1,m}(\alpha, \tilde{q}_{i,j}) - 2m\tilde{q}_{i,j} f_{0,m}(\alpha, \tilde{q}_{i,j}) - m\tilde{q}_{i,j}^2 \frac{df_{0,m}}{d\tilde{q}_{i,j}}(\alpha, \tilde{q}_{i,j}) + 2\tilde{q}_{i,j} \frac{df_{1,m}}{d\tilde{q}_{i,j}}(\alpha, \tilde{q}_{i,j}) \right]}{\frac{p_{1,\alpha}(\tilde{q}_{i,j}/2)}{2} [\ln f_{0,0}(\alpha, \tilde{q}_{i,j}) + 1] + \sum_{m=1}^{+\infty} \frac{df_{0,m}}{d\tilde{q}_{i,j}}(\alpha, \tilde{q}_{i,j}) [\ln f_{0,m}(\alpha, \tilde{q}_{i,j}) + 1]} \ln 2. \quad (33)$$

#### 7.4 Model-based Algorithm

In order to speed the allocation process up, Parisot *et al.* (2003) propose to use some offline computed Look-Up Tables (LUT) [30] to solve the system (32). They propose to exploit two parametric curves:

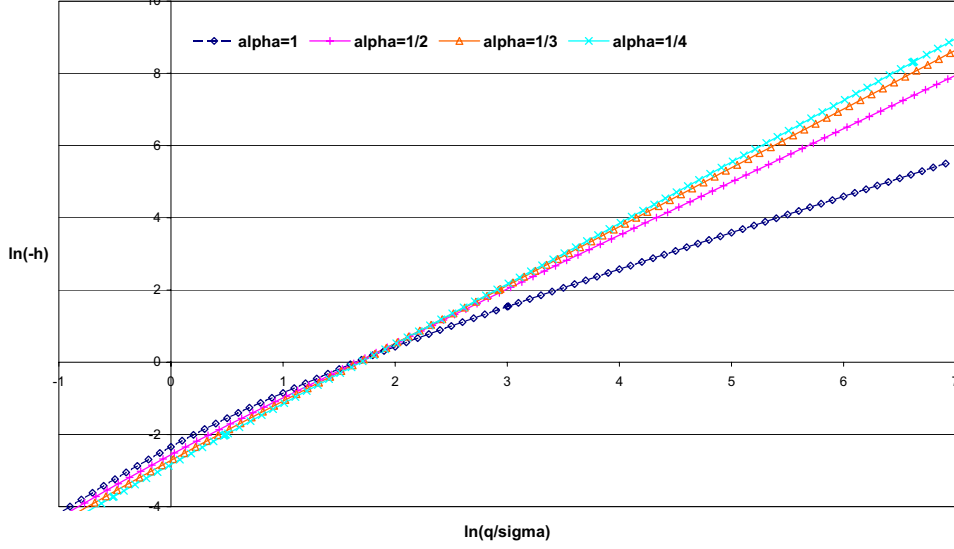


Fig. 6. First LUT used to solve the system (32):  $\ln(-h_\alpha)$  according to  $\ln(\tilde{q})$ , for different  $\alpha$ .

- $[\ln(\tilde{q}); \ln(-h_\alpha)]$ : this LUT allows to compute the quantization step  $q$  corresponding to a specific  $h_\alpha(\tilde{q}_{i,j})$ . Fig. 7.4 shows the parametric curves corresponding to this LUT. This allows to compute the quantization steps verifying (32a).
- $[R; \ln(-h_\alpha)]$ : this LUT allows to compute the bitrate  $R$  corresponding to a specific  $h_\alpha(\tilde{q}_{i,j})$ . Fig. 7.4 shows the parametric curves corresponding to this LUT. This permits to verify the constraint on the bitrate (32b);

In that case, the algorithm given in section 6.3 becomes:

- (1) compute the variance  $\sigma_{i,j}^2$  and the parameter  $\alpha_{i,j}$  for each set  $(i, j)$ ;
- (2) a value of  $\lambda$  is given. For each set  $(i, j)$ , compute  $h_\alpha(\tilde{q}_{i,j})$  thanks to the right-hand side of (32a). Then, use the LUT of  $[R; \ln(-h_\alpha)]$  to compute the corresponding bitrate  $R_{i,j}$ ;
- (3) while (32b) is not verified, calculate a new  $\lambda$  by dichotomy and return to step 2;
- (4) At this step, the optimal  $\lambda$  is known. Thus, for each set  $(i, j)$ , use the LUT of  $[\ln(\tilde{q}); \ln(-h_\alpha)]$  to compute the optimal quantization step  $q_{i,j}$  corresponding to the value of  $h_\alpha(\tilde{q}_{i,j})$  found in step 2.
- (5) stop.

### 7.5 Complexity

In this section, we evaluate the complexity of the model-based algorithm to show the interest of the proposed approach.

Step 1 of the algorithm permits the computation of the variance  $\sigma^2$  and of

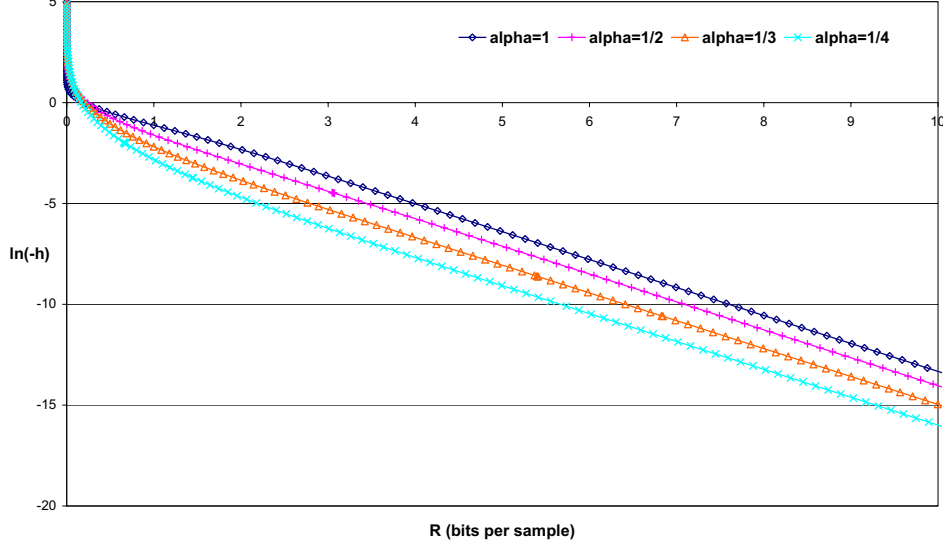


Fig. 7. Second LUT used to solve the system (32) :  $\ln(-h_\alpha)$  according to  $R$ , for different  $\alpha$ .

the parameter  $\alpha$ . The parameter  $\alpha$  is computed from the variance and the fourth-order moment for each component set [31]. This step can be done in 4 operations per component.

At step 2, after the computation of  $\ln(-h_\alpha)$  using  $\lambda$ ,  $\sigma^2$  and Eq. (32a), the set of  $\{R_{i,j}\}$  is computed at low cost by addressing the LUT associated to  $[R; \ln(-h_\alpha)]$ .

Step 3 consists in computing a simple weighted sum of the bitrates estimated at step 2 (2 arithmetic operations per component set) to verify the constraint on the global bitrate. The computation of a new  $\lambda$  is done by a simple dichotomy.

At step 4, the set of quantization steps  $\{q_{i,j}\}$  is computed at low cost by addressing the LUT associated to  $[\ln(\tilde{q}); \ln(-h_\alpha)]$ .

The convergence of the algorithm is reached after few iterations (lower than 5). Finally, the step 1 represents the highest computational cost of this algorithm, with 4 operations per sample, hence a computational complexity of approximately 12 operations per semi-regular vertex. This involves a fast allocation process with a very low computational complexity, taking less than 0.4 second on a Pentium III 1GHz, 512 Mbytes RAM.

## 8 Experimental Results

This section presents some experimental results of the proposed coder, and we compare its performances to some state-of-the-art coders. These coders are:

- The zerotree coder of normal meshes (NMC) [8];
- The EQ mesh coder (EQMC) for normal meshes [10];
- The original Zerotree Coder (PGC) [7] for semi-regular meshes, including the remeshing technique MAPS [3];

The coders NMC and EQMC are currently the most efficient geometry coders. In order to encode the connectivity of the base mesh, we use the topology coder of Touma and Gotsman [25] as in the three state-of-the-art coders previously denoted. This permits to compare only the performances of the different geometry coders.

Fig. 8, 9, and 10 show the resulting PSNR curves according to the bitrate per irregular vertex, for the models HORSE, RABBIT, and VENUS. The PSNR is given by

$$PSNR = 20 \log_{10} \left( \frac{bb}{d_s} \right),$$

where  $bb$  is the bounding box diagonal and  $d_s$  is the surface-to-surface distance between the irregular input mesh and the reconstructed semi-regular one (computed with MESH [27]). We observe that the proposed coder provides better or equal results than the state-of-the-art coders, for any bitrate. We obtain similar results with the models FELINE, SKULL and MOLECULE. This is remarkable since theoretically the MSE is a suitable criterion only in case of optimal rate coding (high bitrates). Finally, we find experimentally that the MSE is always a suitable criterion, for any bitrate range.

Table 3 gives the PSNR values relative to the proposed coder and to the coder NMC according to the global bitrate for all the models. The proposed coder is always better than NMC. In particular, we observe an improvement up to +2.5 dB, in coding performance. In addition, Fig. 11 provides some visual benefits relative to the use of the proposed coder. This figure shows the distribution of the reconstruction error on the object FELINE, quantized with the proposed coder (Fig. 11(a)) and with *NMC* (Fig. 11(b)). The colour corresponds to the magnitude of the distance point-surface normalized by the bounding box diagonal, between the input irregular mesh and the quantized one (computed with MESH [27]). One can argue that *NMC* leads to more local errors than the proposed algorithm. Moreover, Fig. 12 shows renderings of different compressed versions of VENUS. This demonstrates that even at low bitrates the meshes quantized with the proposed algorithm is not so far from the original irregular one.

Finally, we can conclude that in case of normal meshes, the weighted MSE is a suitable criterion to evaluate the reconstruction error between the input irregular mesh and the semi-regular quantized one. This involves an efficient geometry quantization at low computational cost thanks to the model-based bit allocation.

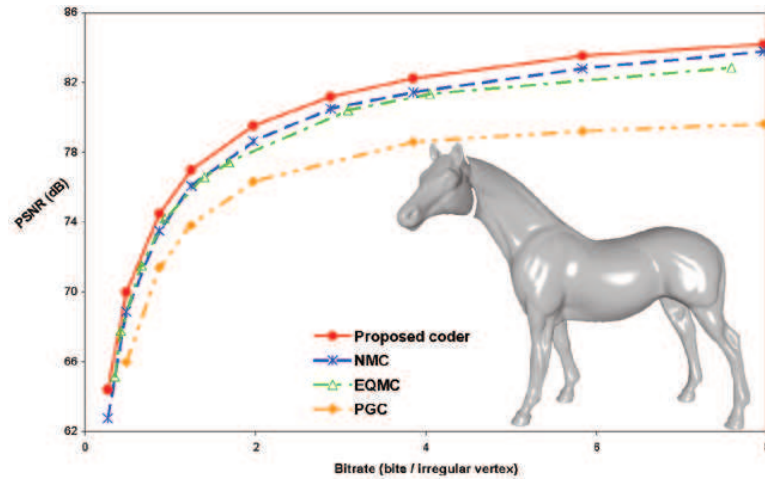


Fig. 8. Bitrate-PSNR curve for the object HORSE.

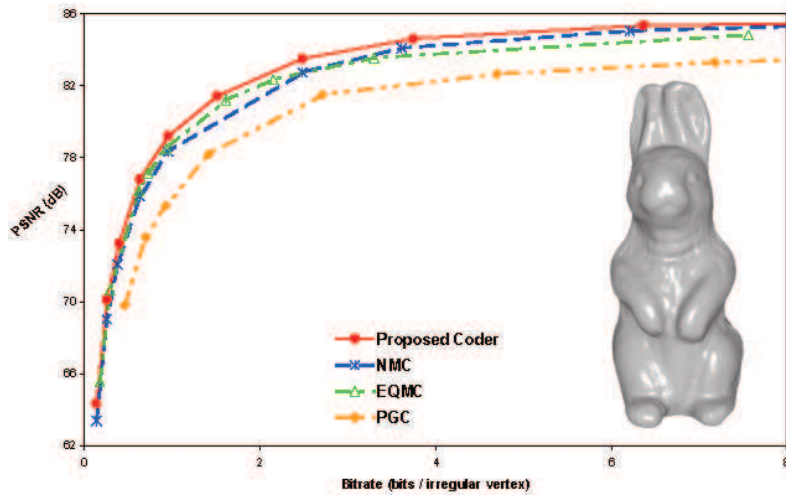


Fig. 9. Bitrate-PSNR curve for the object RABBIT.

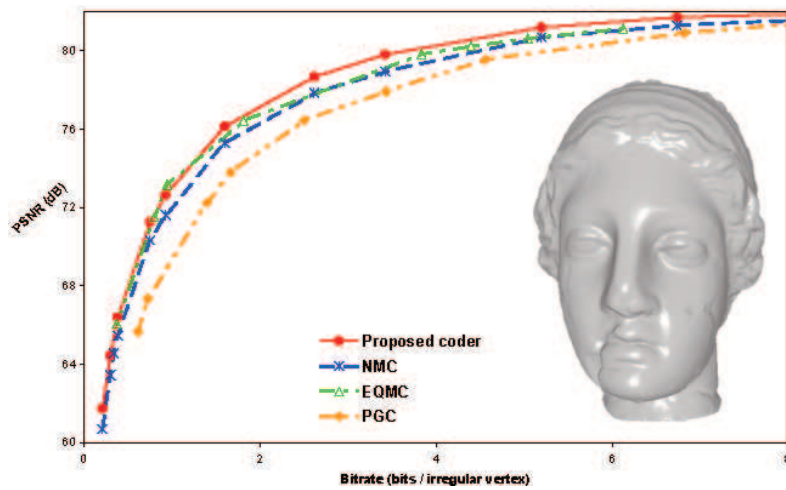


Fig. 10. Bitrate-PSNR curve for the object VENUS.

RABBIT								
Bitrate (bits/iv)	0.14	0.26	0.38	0.92	2.48	3.61	6.20	9.03
Proposed Coder	64.35	70.15	73.26	79.20	83.50	84.61	85.35	85.50
NMC	63.38	69.04	72.09	77.99	82.74	84.08	85.05	85.42
<b>Improvement</b>	<b>0.97</b>	<b>1.11</b>	<b>1.17</b>	<b>1.21</b>	<b>0.76</b>	<b>0.53</b>	<b>0.30</b>	<b>0.07</b>
HORSE								
Bitrate (bits/iv)	0.26	0.87	1.24	1.97	2.88	3.85	5.84	11.51
Proposed Coder	64.39	74.47	76.97	79.51	81.18	82.22	83.52	84.79
NMC	62.75	73.51	76.06	78.63	80.50	81.43	82.80	84.36
<b>Improvement</b>	<b>1.64</b>	<b>0.96</b>	<b>0.91</b>	<b>0.88</b>	<b>0.68</b>	<b>0.79</b>	<b>0.72</b>	<b>0.43</b>
VENUS								
Bitrate (bits/iv)	0.20	0.34	0.92	1.60	3.42	5.20	6.74	9.06
Proposed Coder	64.39	65.64	76.97	79.51	81.18	82.22	83.52	84.79
NMC	60.71	64.59	71.63	75.29	78.95	81.32	81.77	80.69
<b>Improvement</b>	<b>1.03</b>	<b>1.05</b>	<b>1.02</b>	<b>0.86</b>	<b>0.87</b>	<b>0.52</b>	<b>0.41</b>	<b>0.25</b>
FELINE								
Bitrate (bits/iv)	0.39	0.71	1.00	1.30	2.22	3.28	4.37	5.05
Proposed Coder	59.34	65.35	68.10	70.60	73.72	74.84	75.14	75.22
NMC	57.03	62.86	66.57	68.99	72.99	74.47	74.96	75.11
<b>Improvement</b>	<b>2.32</b>	<b>2.49</b>	<b>1.53</b>	<b>1.62</b>	<b>0.75</b>	<b>0.36</b>	<b>0.18</b>	<b>0.11</b>
SKULL								
Bitrate (bits/iv)	0.21	0.53	0.82	1.52	2.25	4.08	5.08	7.90
Proposed Coder	54.60	61.19	64.70	68.38	70.65	72.96	73.38	73.58
NMC	53.75	60.98	63.96	67.73	70.27	72.60	73.19	73.54
<b>Improvement</b>	<b>0.85</b>	<b>0.21</b>	<b>0.74</b>	<b>0.66</b>	<b>0.39</b>	<b>0.36</b>	<b>0.19</b>	<b>0.04</b>
MOLECULE								
Bitrate (bits/iv)	0.43	0.89	1.69	2.25	3.10	6.08	7.60	10.79
Proposed Coder	44.69	52.00	57.55	59.53	61.54	64.30	65.38	66.15
NMC	43.57	50.93	56.30	58.48	60.73	63.43	64.59	65.59
<b>Improvement</b>	<b>1.12</b>	<b>1.07</b>	<b>1.24</b>	<b>1.05</b>	<b>0.81</b>	<b>0.87</b>	<b>0.79</b>	<b>0.55</b>

Table 3

Proposed coder versus the state-of-the-art NMC: PSNR improvement (in dB) for 6 typical models.

## 9 Conclusions

In this paper, we design an original wavelet coder based on the rate-distortion optimization, for densely sampled triangular meshes. This coder includes the *normal remesher* [4], and an original model-based bit allocation that optimizes the quantization of the wavelet coefficients. By assuming that the quantization of the coarser levels does not modify significantly the computation of the local coordinate systems, we argue that the weighted sum of the MSE of quantization of each wavelet component set is a suitable distortion criterion to evaluate the reconstruction error between the irregular input mesh and the

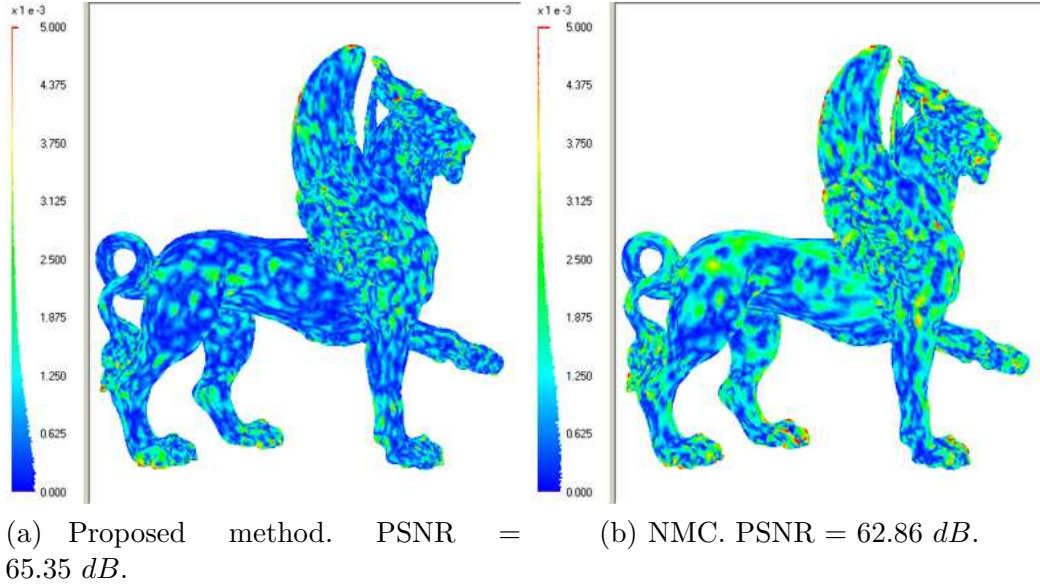


Fig. 11. Distribution of the reconstruction error on the object FELINE. The total bitrate is equal to 0.71 bits/irregular vertex.

reconstructed normal one during the bit allocation. By minimizing this MSE for a user-given target bitrate, the allocation process dispatches the bit budget across the wavelet subbands according to their influence on the quality of the reconstructed mesh. Moreover, the use of theoretical models for the distortion and the bitrate of each component set involves a very fast computation of the optimal quantization steps. Experimental results demonstrate that the proposed approach achieves better results than the two state-of-the-art normal mesh coders [8,10] (up to +2.5 dB at low bitrates), for a very low computational cost.

## Acknowledgements

Datasets are courtesy of Cyberware, Headus, The Scripps Research Institute, and University of Washington. We are particularly grateful to Igor Guskov for providing us with his normal meshes and his executable NMC, and Shridar Lavu for providing us with his executable EQMC. We also want to acknowledge the anonymous reviewers for their advises which permitted to improve the quality of the paper.

## References

- [1] P. Alliez, C. Gotsman, Recent advances in compression of 3D meshes, in: Proceedings of the Symposium on Multiresolution in Geometric Modeling, 2003.



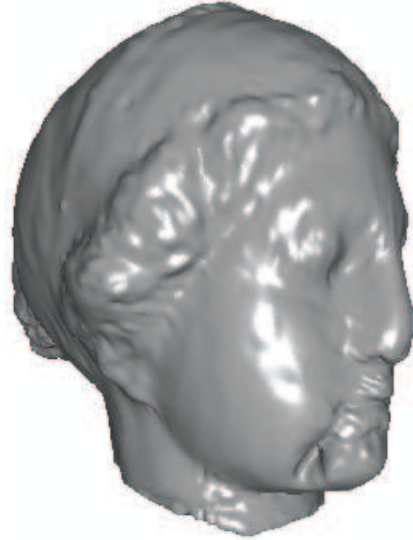
(a) Input mesh.



(b) 2.6 bits/irregular vertex (*Bitstream size*=16340 Bytes).



(c) 0.7 bits/irregular vertex (*Bitstream size*=4677 Bytes).



(d) 0.3 bits/irregular vertex (*Bitstream size*=1842 Bytes).

Fig. 12. Renderings of different compressed versions of VENUS.

- [2] M. Levoy, The digital michelangelo project, in: Proceedings of the 2nd International Conference on 3D Digital Imaging and Modeling, 1999.
- [3] A. Lee, W. Sweldens, P. Schröder, P. Cowsar, D. Dobkin, MAPS: Multiresolution adaptive parameterization of surfaces, in: Proceedings of SIGGRAPH'98, 1998.
- [4] I. Guskov, K. Vidimce, W. Sweldens, P. Schröder, Normal meshes, in: SIGGRAPH 2000, Computer Graphics Proceedings, 2000, pp. 95–102.
- [5] X. Gu, S. J. Gortler, H. Hoppe, Geometry images, in: Proceedings of the



29th annual conference on Computer graphics and interactive techniques, ACM Press, 2002, pp. 355–361.

- [6] A. Lee, H. Moreton, H. Hoppe, Displaced subdivision surfaces, in: Proceedings of ACM SIGGRAPH 2000, 2000, pp. 85–94.
- [7] A. Khodakovsky, P. Schröder, W. Sweldens, Progressive geometry compression, in: Proceedings of SIGGRAPH 2000, 2000.
- [8] A. Khodakovsky, I. Guskov, Normal mesh compression, Geometric Modeling for Scientific Visualization, Springer-Verlag (2002).
- [9] J.-Y. Sim, C.-S. Kim, C. J. Kuo, S.-U. Lee, Normal mesh compression based on rate-distortion optimization, in: Proceedings of the international Workshop on MultiMedia Signal Processing, 2002.
- [10] S. Lavu, H. Choi, R. Baraniuk, Geometry compression of normal meshes using rate-distortion algorithms, in: Proceedings of the Eurographics/ACM SIGGRAPH symposium on Geometry processing, 2003.
- [11] S. Lopresto, K. Ramchandran, M. Orchard, Image coding based on mixture modeling of wavelet coefficients and a fast estimation-quantization framework, in: Proceedings of IEEE Data Compression Conference (DCC), Snowbird, Utah, 1997, pp. 221–226.
- [12] F. Payan, M. Antonini, Weighted bit allocation for multiresolution 3D mesh geometry compression, in: Proceedings of SPIE Visual Communications and Image Processing (VCIP) Conference, Lugano, Switzerland, 2003.
- [13] F. Payan, M. Antonini, 3D multiresolution context-based coding for geometry compression, in: Proceedings of IEEE International Conference in Image Processing (ICIP), Barcelona, Spain, 2003.
- [14] M. Chow, Optimized geometry compression for real-time rendering, in: Proceedings of the 8th conference on Visualization '97, IEEE Computer Society Press, 1997, pp. 347–354.
- [15] J. Li, J. Li, C. C. Kuo, Progressive compression of 3D graphic models, in: International Conference on Multimedia Computing and Systems, 1997, pp. 135–142.
- [16] D. King, J. Rossignac, Optimal bit allocation in 3D compression, Journal of Computational Geometry, Theory and Applications 14 (1-3) (1999) 99–118.
- [17] Z. Karni, C. Gotsman, Spectral compression of mesh geometry, In ACM SIGGRAPH Conference Proceedings (2000) 279–286.
- [18] O. Sorkine, D. Cohen-Or, S. Toledo, High-pass quantization for mesh encoding, in: Proceedings of the Eurographics/ACM SIGGRAPH symposium on Geometry processing, Eurographics Association, 2003, pp. 42–51.
- [19] D. Luebcke, B. Halle, Perceptually driven simplification for interactive rendering, in: Proceedings of the 12th Eurographics Workshop on Rendering Techniques, 2001.

- [20] P. Cignoni, C. Rocchini, R. Scopigno, Metro: Measuring error on simplified surfaces, *Computer graphics Forum* 2 (17) (1998) 167–174.
- [21] C. Gotsman, S. Gumhold, L. Kobbelt, Simplification and compression of 3D meshes, in: M. F. A. Iske, E. Quak (Ed.), *In Tutorials on Multiresolution in Geometric Modelling (Munich Summer School Lecture Notes)*, 2002.
- [22] D. Zorin, P. Schröder, W. Sweldens, Interactive multiresolution mesh editing, *Computer Graphics* 31 (Annual Conference Series) (1997) 259–268.
- [23] W. Sweldens, The lifting scheme: A construction of second generation wavelets, *SIAM Journal on Mathematical Analysis* 29 (2) (1998) 511–546.
- [24] P. Schröder, W. Sweldens, Spherical wavelets: Efficiently representing functions on the sphere, *Proceedings of SIGGRAPH 95* (1995) 161–172.
- [25] C. Touma, C. Gotsman, Triangle mesh compression, *Graphics Interface'98* (1998) 26–34.
- [26] A. Gersho, Asymptotically optimal block quantization, *IEEE Transactions on Information Theory* (25) (1979) 373–380.
- [27] N. Aspert, D. Santa-Cruz, T. Ebrahimi, Mesh: Measuring errors between surfaces using the hausdorff distance, in: *Proceedings of the IEEE ICME, Vol. I*, 2002, pp. 705 – 708.
- [28] B. Usevitch, Optimal bit allocation for biorthogonal wavelet coding, in: *IEEE Data Compression Conference*, 1996.
- [29] F. Payan, M. Antonini, Mean square error for biorthogonal M-channel wavelet coder, *IEEE Transactions on Image Processing* (second review), 2004.
- [30] C. Parisot, M. Antonini, M. Barlaud, 3D scan based wavelet transform and quality control for video coding, *EURASIP journal on Applied Signal Processing*, Special issue on multimedia Signal Processing, January, 2003.
- [31] J. Kasner, M. Marcellin, B. Hunt, Universal trellis coded quantization, *IEEE Transactions on Image Processing* 8 (12) (1999) 1677–1687.
- [32] F. Payan, M. Antonini, Multiresolution 3D mesh compression, in: *Proceedings of IEEE International Conference in Image Processing (ICIP)*, Rochester, USA, 2002.